View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by AIS Electronic Library (AISeL)

Association for Information Systems

AIS Electronic Library (AISeL)

ACIS 2013 Proceedings

Australasian (ACIS)

2013

SYSTEMIC POWER, DISCIPLINARY AGENCY, AND DEVELOPER–BUSINESS CLIENT RELATIONS

Bruce Rowlands
Griffith University, b.rowlands@griffith.edu.au

Karlheinz Kautz
University of Wollongong, karlheinz.kautz@rmit.edu.au

Follow this and additional works at: https://aisel.aisnet.org/acis2013

Recommended Citation

Rowlands, Bruce and Kautz, Karlheinz, "SYSTEMIC POWER, DISCIPLINARY AGENCY, AND DEVELOPER–BUSINESS CLIENT RELATIONS" (2013). *ACIS 2013 Proceedings*. 123.
https://aisel.aisnet.org/acis2013/123

This material is brought to you by the Australasian (ACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ACIS 2013 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Information Systems: Transforming the Future

24th Australasian Conference on Information Systems, 4-6 December 2013, Melbourne

Proudly sponsored by

ACIS 2013 Principal Sponsor

Systemic Power, Disciplinary Agency, and Developer–Business Client Relations

Bruce Rowlands
School of Information & Communication Technology
Griffith University
Brisbane, Australia
Email: b.rowlands@griffith.edu.au

Karlheinz Kautz
Faculty of Business
University of Wollongong
Wollongong, Australia
Email: Kautz@uow.edu.au

Abstract

This paper presents Hardy's multi-dimensional model of power and illustrates its application to the field of IS. Findings from a case study of developer—business client power relations within a large financial institution are presented. Our findings indicate that from the developers' perspective, the client exercised near complete control, with developers unwittingly playing a cooperative but submissive role. Our study makes two principal contributions. First, we combine Hardy's (1996) multi-dimensional power framework and the principles of Pickering's (1995) version of disciplinary agency to propose why the developer was compliant in this scenario of power inequality. Second, we examine how a development methodology helped convey symbolic and disciplinary power. By doing so we gain rich insight into how meaning power, and the power of the system institutionalised within the methodology, can constrain the actions of developers.

Keywords: Systemic power, disciplinary agency, method enactment, developers, qualitative research.

INTRODUCTION

This paper reports on field research into the relations between developers and the business client, describes a case scenario where developers perceive the systems development process as unequal, and explores the role that a systems development methodology (SDM) can play in influencing this relationship. In our field interviews of developers across all levels of the IT division within a major international bank, the case shows from the perspective of developers the business client exercising nearly complete control over the development process and systems developers playing a cooperative, but submissive role. This situation is unusual and is in direct contrast to the majority of findings reported in the literature. Our focus in this paper is on understanding and explaining this unusual case based solely on developers' perceptions and views.

We begin our paper by developing a theoretical framework to explore the relationship and complexity of dynamics between developers and the business client. Key to this framework is a theory of power relations first articulated by Lukes (1974), and adapted by Hardy (1985, 1996) and her colleagues (Hardy and Leiba-O'Sulivan, 1998), known as a theory of unobtrusive power. This theory highlights the influence of the institutionalised context on SDM enactment – termed the 'power of the system' or systemic power. This dimension of power and its influence has not been well discussed or examples given in the IS literature to date.

Our paper also builds on recent work concerning developers and business clients as political actors (Howcroft and Wilson, 2003; Markus and Mao, 2004; Hekkala, Urquhart and Iivari, 2009; Dhillon, Caldeira and Wenger, 2011). However, unlike prior studies which have primarily focussed on mechanisms of control from the business client perspective (Sauer and Lau, 1997; Heiskanen, Newman and Eklin, 2008), we only present and focus on the perspective of the developer. Our task is to addresses the following research question: why was the developer compliant in this particular scenario of power inequality?

Finally, three key terms as used in this paper need defining. Our definition of the **business client** includes all those business client departments who pay for, operate and interact with software systems to achieve organisational goals. The business client generally initiates the development or enhancement of systems but most importantly, funds the development of these systems. Although the business client is normally not the end-user (Kautz, 2011), in this case for reasons of simplicity we call both stakeholder groups the *client*. From Markus and Bjørn-Anderson (1987) we understand **systems developers** within *The Bank* to include all those individuals and groups, both inside and outside *The Bank*, who engage with the business client; that is, assess their needs, propose

solutions, and develop software systems. This paper also uses the term **systems development methodology** very broadly along with Fitzgerald, Russo and Stolterman (2002) as any formally documented in-house or commercially available systems development approach.

LITERATURE: POWER AND CONFLICT IN INFORMATION SYSTEMS DEVELOPMENT

We begin our review claiming that there are too few studies illustrating the SDM enactment process in its social, political and organisational contexts (Kautz, Madsen and Nørbjerg, 2007); and an imbalance of research about the motivations and actions of developers within the developer-business client relationship. Markus and Mao (2004) identified that the voice of the developer compared to those of the business client has been considerably under-researched. However, the relationship between power and information systems development has been extensively studied (Sillince and Mouaket, 1997; Jasperson, Carte, Saunders, Butler, Croes, and Zheng, 2002; Silva, 2007; Sabherwal and Grover, 2009).

Within the field of IS, information and power were considered to be synonymous, and hence those who built systems were viewed as instrumental in influencing power relationships (Markus and Bjørn-Anderson, 1987; Beath and Orlikowski (1994). Our review of the literature portrays developers as being in control. Smith (1990) found that a significant feature of the relationship between business clients and developers, is that the antagonism appears to be one-sided — from business clients towards IS professionals. Beath and Orlikowski (1994) identified a contradictory dichotomy between business clients and developers, with developers in command and business clients submissive, while business clients were expected to be responsible for the outcomes of the development process. In more recent findings, Hussain and Cornelius (2009) reported on how domination structures favouring IT management were produced and authority over business clients was secured. However, in contrast, our case will indicate that the policies of the business client institutionalised in the SDM constitute a covert exercise of power, and gave the client power by default over developers.

In understanding our case it is difficult to determine *a priori* how a SDM might influence the relationship between the business client and the developer because there is no well established theory specific to systems development methodologies, or 'widely accepted framework for studying the use of these methods' (Fitzgerald, Russo and Stolterman, 2002:12). Only a few studies focus on the deployment of SDMs in their social and organisational contexts and the power relations existing between developers and the business client in this context. Sauer and Lau (1997) reported on a case where the role of the SDM was highly influential in distributing the power relations between business client and developer and where the business client was able to impose their priorities demonstrating overt power in the systems development process. These findings recognised that business clients are a source of influence on methodology enactment because they controlled the resources. Nandhakumar and Avison (1999) highlighted various influences such as developers' knowledge about methodologies, implicit social norms, organisational form, and culture. Madsen et al's (2006) conceptualisation portrayed the role and usefulness of methodologies as a means for communication, coordination and (re)direction, rather than as a rigorous or rigid means for control. Whereas Huisman and Iivari (2006) in their study of the difference in perception between IS managers and developers about the deployment of SDMs found that both groups saw SDMs as a control technology in terms of keeping to deadlines and budget, yet they offered no discussion of how control was achieved.

Our review suggests that no-one has studied a situation that we have encountered where developers perceive the business client to exert power over developers both overtly and by default covertly in the systems development process through the enactment of a SDM. In the context of understanding our case scenario, the literature is limited. What is unique in this paper is that prior SDM enactment studies have not recognised cases where covert structures such as symbolic and systemic power embedded within the accepted and every-day use of the SDM directly determine power relationships between the developer and the client.

THEORETICAL FRAMEWORK

In this case, we provide an argument that SDMs are institutions that exert their own form of disciplinary agency (Pickering, 1995). Pickering argues that the agency of a discipline – such as information systems development – leads people through a series of actions and also neutralises these actions for them. In Section 5 *Analysis of the Case* we provide a grounded description of systems developers working within a discipline that provides scaffolding for their actions. Through the application of Hardy's (1996) concept of systemic power, we offer plausible explanations of how the discipline of systems development through accepted knowledge that everyone takes for granted ensures that working relations between client and developer involving the SDM are translated into routine organisational work practices that become unquestioned normative structures of the organisation.

The Concept of Power

Power is a complex concept where any consideration of the role of power depends on which perspective of power one chooses (Sillince and Mouakket, 1997). In the mainstream organisational literature, power has been conceptualised as the mobilisation of resources such as money, funding, the ability to hire and fire, and to influence problem solving (Hardy, 1985). Power, in this case, involves the possession of strategic resources on which others depend, and enables actors to produce the substantive outcomes they desire. This type of power has been referred to as 'overt power' (Hardy, 1985) or 'episodic power' (Clegg, 1989). Episodic power refers to that which is enacted in relatively discrete, strategic events that are initiated by self-interested actors (Clegg, 1989). This mode of power has dominated the study of power in organisations and has been approached from a wide variety of perspectives (Pfeffer, 1981). Overt or episodic power can take the form of economic power (power to reward/punish actors for outcomes based on funding), or legitimate power (power grounded in the hierarchical position of individuals). This view of power is also person based, meaning that power is located within the individual, also known as an agency view of power.

In the organisational literature, there are views that both challenge and go beyond the overt or an agency view of power. For example, power can be understood as exercised in relations between people. In this relational view, power is not seen as being in any one place or as something people have, but is dispersed and enacted through the range of relational interactions between people (Horton, 2003).

In contrast to episodic forms of power, relational forms of power work through the routine, ongoing practices of organisations to advantage particular groups without those groups being obviously or clearly connected to the establishment or maintenance of those practices (Lawrence, Winn, and Jennings, 2001) . This stream of research has emphasised that power is not always overt, it is sometimes much more subtle and often invisible. This stream includes processes of meaning creation and persuasion, and represents in Hardy's (1985) terms an 'unobtrusive' form of power as it creates legitimacy and justification for certain arrangements or actions so that they are not questioned. This type of unobtrusive or covert power is not predetermined by the control of resources; rather it emerges during actors' interactions and relationships and is the outcome of the translation of meaning and a socialisation process occurring within the IT industry and at the workplace.

Hardy's Multi-Dimensional Model of Power

Hardy (1985; 1996) integrated Lukes' (1974) view of power into a four-dimensional model which incorporates both the use of overt power to defeat declared and identifiable opponents; and its use to prevent resistance, known as covert or *unobtrusive power*. Unobtrusive power concerns attempts to create legitimacy and justification for certain arrangements by the powerful, so that outcomes are never questioned by the powerless. Following Hardy (1985), Hardy (1996) and Hardy and Leiba-O'Sullivan (1998) we present an organisational power framework that operates along four dimensions.

Hardy's (1985) first dimension known as the *Power of Resources* seeks to study actual behaviour with the locus of power being presumed to reside with the victor in a decision situation involving a conflict of interest. At this level the focus is on the use of resources and decision outcomes, and helps to explain decision outcomes as political rather than rational (Hardy and Leiba-O'Sullivan, 1998). The second dimension known as the *Power of Decision-making Processes* reveals the ways in which some groups may dominate others in relation to decision making by considering ways in which decisions are prevented from being taken on potential issues over which there is an observable conflict of interest. The third dimension known as the *Power of Managing Meaning* addresses exercises of power that prevent potential issues from arising by considering the many ways through which potential issues are kept out of politics, whether through the operation of social forces and institutional practices or through individuals' decisions. The third dimension relates to the management of meaning or the manipulation of perception through the use of symbols, rituals, language and myths. The main contribution of the third dimension is to move thinking about a concept of power beyond a link with conflict, and to understand how issues can be prevented from arising at all (Horton 2003). The fourth dimension known as the *Power of the Organisational System* or systemic power, is characterised by the unconscious acceptance of values, traditions and structures of a given institution or society. According to Hardy, power has many unintentional effects, is pervasive and has an invisible meaning.

From our search of the information systems literature, Hardy's first three dimensions of power framework have used in prior IS studies (Dhillon, 2004; Howcroft and Light, 2006; Howcroft and McDonald, 2007; and Dhillon et al, 2011); whereas applications of Hardy's 4th dimension of power remains almost absent in the IS literature (Dhillon et al, 2011).

RESEARCH APPROACH

The research approach adopted in this study is that of an interpretive case study (Walsham, 1995; Klein and Myers, 1999). How systems developers interpret an SDM and their relationship with the business client is

important because those with different interpretations will enact the SDM differently. Eisenhardt (1989) advocated the use of a single site case as being appropriate. Our approach was exploratory and was significantly shaped by the interplay between theory and research material. However, our aim to analyse the developer-business client relationship from a power perspective was not initially clearly formulated. During initial coding and data analysis, writing, discussion, review and re-writing, this focus however did become clearer.

The selection of the case site was based on a combination of accessibility to the company's IT managers and project members, and interestingness – in the sense that the chosen bank is one of Australia's top four banks, and its IT organisation is considered to be a leading player in providing state-of-the-art IS solutions to customers.

The sampling strategy for the interviews included a combination of purposeful and theoretical sampling (Schwandt, 2001). Three occupational functions within *The Bank* were selected for their similarities as well as their differences. Interviews were only conducted with systems developers comprised of project managers, senior consultants, and consultants within the systems support, new development, and method support sub-divisions of the IT division. A total of thirty interviews were conducted with twenty-eight informants from different projects and at varying levels within the organisation (c.f. Table 1). Two method support personnel were interviewed twice. In the majority of cases, each face to face interview was complemented by a follow-up email to clarify issues and to obtain supplementary information.

Table 1: Number of Interviews by Job Role within the Bank, and Project Type

		Job Role		
Project Type	Project Manager	Senior Consultant	Consultant	Total
System Support	2	8	6	16
New Development	2	4	3	9
Method Support	3	2	0	5
Total	7	14	9	30

As previously stated, we did not set out to study control or power, instead we deliberately kept the case interview questions open leaving the developers to tell us their story about what influenced them in their use of the SDM. Through coding and drawing meaning from the interviews, a recurring theme among the responses were developers describing their subordinate relationship when dealing with the business client. This key theme turned our attention towards control and the authority relations existing between the business client and developers. We then considered the relations between developers and the business client as a form of power play, operating essentially within a political arena.

Following data collection and initial analysis, the author developed and shared a case report, including a case summary and preliminary elements of analysis, with a current project manager involved in the use of the SDM. This manager commented on the report and gave confirmation of many points and qualifications of others. Drawing on this report, the interviews, the scrutiny of informants, and the researcher's subjective understanding of the case phenomena, the paper presents the analytical case.

ANALYSIS OF THE CASE

Context

The Bank is large in terms of Australian corporation size and is old, traditionally stable and bureaucratic. *The Bank's* IT division consists of approximately 700 people, half of whom work in application support. Each new development or support team had a project manager who reported to a business unit department manager who had overall control of the project through budget and a stage-gate funding approval process. The size and composition of project teams usually consisted of core people on the project e.g. project manager, business analyst, solution designer, developers and sub-providers; and the other teams that interface with this application: business business clients, hardware vendors, the telecommunications provider, and other partners who may be involved in outsourced business processes.

The SDM and Practice

To develop, customise or maintain these systems, the IT division had developed and documented an internal SDM applicable for all development and maintenance tasks. The methodology was based on traditional 'waterfall' lifecycle phases. The use in practice of the methodology was mandatory and covered all new development, package acquisitions and any planned changes to existing systems, except urgent fixes. For many developers, knowledge of the SDM was acquired on the job and internalised over time. In reality, method use was an unconscious process involving tacit knowledge being inter-twined with practical experience over time; which according to Hardy (1996) is a form of systemic power.

Projects are initiated by the client in most instances. The client may wish to create a new product, a system, or change existing systems to comply with regulations. This initiation happens within a team in a business unit. The business team engages with their management to get approval and funding, and if approved, a project manager is appointed and the project commences with the business client securing IT services from a supplier in most cases the in-house IT division. However, business has the option to procure software services from external third parties, representing in Hardy's (1996) terms a form of resource power.

An emphasis of the SDM within each phase and sub-phase of the development lifecycle is the production of documentation. Once a document is produced it is sent for a review to the distribution list identified by the project manager. Key project members, notably the business client, are needed to sign-off or review the documents. Often there are formal document review sessions called "walk-throughs" where key project members from the business client, the hardware vendor, and developers gather with the author of the document from the development team to 'walk through the document' before the sign-off deadline date, which according to Hardy (1996) is a form of decision-making power.

As part of the project management process, the client mandates that all projects are funded in phases, and that some important documents are required as completed deliverables as proof to gain funding for the next stage.

Through empirically supported examples, we illustrate how each dimension of Hardy's framework manifests itself in the day-to-day actions of methodology enactment within *The Bank*.

Dimension 1: Power of Resources, and Dimension 2: Power of Decision Making

In *The Bank*, the SDM was seen by developers as a vehicle to bring project members together and to coordinate their tasks when interacting with business clients and industry bodies. In its most visible form, the SDM mandated documentation throughout all phases of the development lifecycle and required project managers to call for meetings, both formal and informal among affiliates to review project status and sign-off on design documentation. The main reason cited for documenting and seeking to communicate in legitimate ways was to get documents signed and gain approval from the client to commence the next stage of development. To do so, there needed to be visibility of development work as a project manager commented:

> … producing a document is one way of providing visibility of what's actually happening and with all the formal documents that are required to be signed ... its approval to go to the next stage and that the work can be done. Also with funding - project funding is dependent on these documents having to be produced. In a large scale development the funding is very important in each phase. So therefore you have to produce some kind of deliverable to prove what you've done.

The client had the ability to procure services in-house or external to the organisation, and therefore developers are reliant on the client for funding of projects. Secondly, developers couldn't proceed until each stage was signed off as the SDM mandates signatures from business clients and other development partners. Developers viewed sign-off in two ways: positively – to gain approval so that work could commence on the next stage; and negatively as a way whereby business clients maintain power and control over the development process though a decision-making process. A project manager described a functional role of the methodology was for the client to keep control of the project:

> …you have to get sign off at various points. Yes, the methodology is used by the technology people to build things. But, before you can get funding for the next stage the technology group needs to provide to business things for the project to then proceed to the next phase. So if you want funding to go on further, you'll need to do things. So it [the SDM] forces you to do things [produce documentation].

A specific comment was made by the method support manager about the unequal power relationship between the client (addressed as business in the excerpt) and developers:

> Business sometimes do hold development to ransom, so to speak. That's another part of the culture. Really they should be working together to deliver solutions rather than using contracts as ransom to force them to do something. So it ends up, at the end of the day, a lot of management is structured such that project managers and CIO's are rewarded or punished based on their ability to deliver projects on time.

A relative new-comer to *the Bank* (a developer) remarked about how he saw the power relations between project managers and CIO's being rewarded within *the Bank* based on delivering projects on time:

> a lot of the management is structured such that project managers and CIO's are rewarded or punished based on their ability to deliver. So they apply that pressure downward. Business are probably the most in-flexible areas because they're very much used to having a lot of control. They have a lot of power, because they hold the money.

The same developer described the power relationship between the client (again addressed as business in the excerpt) and developers as uneven.

> This organisation has got a really lean cost model. We [developers] all hate it because it's a means where we can get shafted. The business likes it though.

In sum, gaining sign-off was a work structure imposed by the methodology and while not favored by developers, the client maintains strict compliance to these rules. The client was able to exercise overt decision-making power through the generation of specifications requiring sign-off at each stage.

Additionally developers regularly worked with external organisations when aspects of projects had been outsourced, or when dealing with contractors brought in on a needs basis. These interactions brought about change and through this change new work practices shaped how the methodology was used, and in particular, how the SDM evolved over time. In terms of making changes to the methodology, an analyst when describing how his project team was trying to update the methodology to incorporate object-oriented procedures, found resistance from the business client:

> We attempted to modify the requirement specification to make them more like RUP. We wanted to include some of the RUP in the templates. But we had to check it with the business. And then business came back and said they didn't like some of the concepts, like use case diagrams. So we had to remove some of those key parts because business found them confusing which is crazy, as it's a simple concept.

What is interesting in the above excerpt, even though it is clear that the developers were simply trying to update the methodology, is the rejection of the alien nature of the diagrams by the business client adding another layer of frustration to the developer. But, what is more important is that the client had the power of decision making and the power to say 'no'. These transcripts also identified a desire from developers to accommodate an alternative to the traditional systems development lifecycle (SDLC) and to modify the SDM to incorporate new development techniques. This situation provided an example of an apparent conflict of interest in which the client appeared to maintain control over the methodology, to the relative disadvantage of developers being constrained in their work practices by not using modern development techniques, and having to conform to the dictates of the methodology.

Similarly, there was a distinct feeling among developers that the existing SDM was due for an overhaul. Asked what would be required to introduce an entirely new methodology with the existing SDM phased out, the method support manager replied:

> .. it'd be a big effort. However, when I talk to developers they're quite happy to use other processes. And even some of the project managers would be quite happy to move away from the existing methodology. … There seems to be a desire [among developers] to look at something else, rather than what we currently use as our methodology.

However, to introduce a new SDM would involve gaining business agreement. Business would have to be convinced as the method support manager commented:

> This is a business and the methodology is part of a control mechanism. .. we need to show people [business]that there are better things out there, if there are, and will enable them [business] to not only get the controls that they do currently have but be more productive and more appropriate for the way that they do their work.

For developers to replace or de-institutionalise the SDM, they would have to alter the existing power arrangements currently in place. However, as business virtually owned the SDM they exercised all the power in this decision-making process and developers had no way to get this issue on the agenda.

Another interesting observation throughout the excerpts was how the development side referred to the business client. The dichotomy is evident in many of the developer's references to the client as "business" rather than as, for example, "clients", "partners", or "domain experts". The term "business" as used by developers implies superiority: one who consumes, controls, decides, and manages. From the perspective of developers, power was seen to be vested with the client because they controlled the funding. Asked specifically who drives systems development, a senior analyst responded in a way that was representative of many similar comments:

> It's the business, definitely. Sometimes the IT areas will, once they get a project, try to drive what they think. But on the whole, business are paying [for systems] and whatever they want, gets done.

The same analyst even admitted that some project managers are scared of the power that business wields:

> I have worked for managers where they have agreed to deadlines that are too close and not reasonable. Sometimes too, I think they're scared to tell them that their request isn't reasonable.

The above transcripts confirm the inherent power of the client. The interviews are saying that in the end it is the client who has control over resource and decision-making regarding the SDM, and that developers have a subordinate relationship when dealing with the client. Based on what developers told us, and based on developer's assumptions about client views, the advantages in terms of whose interests are met in the systems

development process were clearly in favour of the client.

However, at this level of analysis, the excerpts do not explain why developers were accepting of their situation in relation to an unequal relationship with the client, why demands were not made, and why conflict did not arise. Hardy's 3rd and 4th dimensions of power propose that quiescence on the part of developers may be the result of an unobtrusive exercise of power. According to Lukes (1974) and Hardy (1985) power can be used to prevent people from having grievances by shaping their perceptions and preferences in such a way that they accept their role in the existing order of things, see it as natural, or see no alternative to it. This acceptance on the part of developers explains why there are few direct excerpts in the following sections making reference to the *power of meaning* or to the *power of the system*. Only Hardy's 3rd and 4th power dimensions draw our attention to the unobtrusive exercise of power.

Dimension 3: Power of Meaning

According to Hardy (1985) unobtrusive power is derived from symbolic sources which are brought into play to legitimise outcomes in a process called the 'management of meaning'. In terms of our case, the meaning of the SDM for developers based around its accepted and everyday use − that defined their identity and their relationship with the client – according to Hardy (1996) would obviate any need for more direct forms of control from the client.

Our analysis will show that developers were influenced by symbolic aspects of power – the use of language, symbols and rituals in the workforce. For example, developers valued standardised terms (language) enabling communication of ideas between themselves, the business client and consultants external to the organisation; the habitual use of the SDM in producing lifecycle documentation (symbols) as evidence of design output and work performance; and the use of "walk-through" meetings (rituals) with business clients to validate the 'accuracy' of design decisions to gain their signature of approval.

As a major form of interaction, the following excerpts illustrate where developers sought to communicate in legitimate ways. The interviews indicated that the SDM provided a terminology, so when developers talked with clients and among themselves they knew the language of the business. In other words, the methodology provided a common language, was consistent, and many developers spoke positively in terms of how the methodology facilitated communication within *The Bank*. The following excerpt illustrates:

> Well, you need to have some methodology…you need a vehicle for communicating particularly between the teams. So, if we were working with another group within the organisation or even external to the organisation you have to have a common language. So it's not bad in that respect.

In its most visible form, the SDM mandated documentation throughout all phases of the development lifecycle, and required project leaders to call for meetings, both formal & informal among affiliates to review project status and sign-off on documentation. The methodology in this sense was seen to facilitate project coordination and as a mechanism to review progress.

Producing documents, getting approval and sign-off to commence the next stage was seen as a major functional component of the methodology, and a major reason for having a formal software development process, as one project manager described:

> before you go to the next phase you have to go through the proper approval process again. Usually its a form – you know, fill out this form. You're to say how much money you've spent so far… should we proceed? What is the estimated cost, what is the saving you're going to get, etc. So these are the kind of steps we have to do at the end of each phase. And without getting a formal approval, I couldn't proceed.

Yet, another project manager mentioned a functional role of the methodology through sign-off was for the business client to keep financial control of the project:

> …you have to get sign off at various points. Yes, the methodology is used by the technology people to build things. But before you can get funding for the next stage the technology group needs to provide to the business things for the project to then proceed to the next phase. So, if you want funding to go on further, you'll need to do things. So it forces you to produce documentation, and to seek sign-off.

Producing documentation and seeking sign-off was a work structure imposed by the business client and materialized through the SDM. The methodology mandated the generation of specifications becoming symbols of design output and work performance requiring sign-off at "walk-through" meetings. For example, many newer staff, such as the following junior support programmer, had little or no training in the SDM but was happy to use it. He remarked that using the SDM helped him to appear that he was performing appropriately in the workplace in terms of his relations with others. He admitted:

> I've been using the same documents since I started. I guess [the methodology] has been around for a lot longer than that. Where those documents originated from, I have no idea. So, well, I guess what I'm trying to say is that people are happy with what I deliver.

Using the methodology competently constructed the developers' identity, legitimised their role, and symbolically constructed perceptions that they are professional. Through sign-off and "walk-through" meetings developers were portrayed as dependent on the client to validate and legitimise their contributions to the organisation. In short, identities were constructed through the process of interaction with the business clients' imposed method. A project leader stated it emphatically when she said:

> If I didn't do it [follow the steps and produce documentation] then I wouldn't be doing my job properly.

In terms of symbolism and meaning, the SDM stands for something more than a 'way to build systems'. The meaning of the SDM comes from its context and enactment within the IT division. Analysis in dimensions 1, 2 and 3 show that many developers held the view that the SDM provided a common language and valued standardised terms enabling communication of ideas between developers, business clients, and those external to the organisation. Developers when they joined *The Bank* accepted their role in the existing order of things because they saw it as natural, and the use of the SDM went relatively unnoticed by developers. The SDM has become habitualised and part of the work culture of the organisation. Using the SDM also created an image for developers that they are professional. The SDM helped define developers' identity as competent, legitimised their role as professional and created a positive image controlling their perception. This identity was used to project an individual's legitimacy to the business client. A support programmer commented that the methodology helped define his legitimacy as a professional IT worker. He felt comfortable and others were happy with his work:

> for me personally I'm comfortable with the way I do my work and people I work with haven't had a problem with it. So I guess, approach wise, I'm doing the right thing. And in terms of compliance with the methodology, I haven't hit a problem with that in my years of working.

Cumulatively these excerpts identify norms (attendance at meetings) and values (performance as a professional) that are perceived by developers as natural and legitimate development practice. Unobtrusively, these symbolic aspects of producing documentation, gaining sign-off, and attending "walk-through" meetings are seen by developers as legitimate development policy. Developers did not work outside this policy because they saw it as natural, acceptable, and contextually and culturally grounded. Consequently, developers comply with these work arrangements because it meets their sense of professional reality. In this case, according to Hardy (1985), it is the client who unwittingly manipulated the perception of the SDM in the workplace and therefore the shared meaning of the development process in a process called the 'management of meaning', and consequently was able to exercise symbolic power over developers.

Dimension 4: Power of the System

According to Hardy and Leiba-O'Sullivan (1998) power is embedded in the very fabric of the *system*; it constrains what developers see, how developers see, and limits their capacity for resistance. In this case, the discipline of systems development provided a way of thinking and acting for developers who took for granted the role conferred on the SDM as the accepted way of doing IT work.

Our analysis has shown that developers were constrained by the discipline of systems development that everyone took for granted. For example, the power of the system comprised the taken for granted assumptions and beliefs concerning reporting to the business client, attending walk-through meetings, gaining sign-off, following a lock-step approach to development, and yielding to unreasonable demands based on securing funding for projects. All these form part of the discipline of development work. The paper argues that these assumptions and beliefs about appropriate development practice are reinforced by a combination of the developers' education, work experiences, and work culture within *The Bank*.

To provide further concrete examples illustrating the power of the system, we need to focus outside the organisation so we can better understand the institutionalised and disciplinary practices taking place within *The Bank*. The discipline of systems development in terms of adherence to industry-wide work practices exerted technical standards on *The Bank* and played a significant guiding role as one project manager put it:

> With a mixture of skill sets [in-house and from contractors] it's important that we all have the same standards and the outcomes are the same. A common benchmark is good, and it can be used as a guideline – and the methodology allows everyone to follow a particular guideline.

Even a Java developer who was not keen on the traditional lifecycle of the SDM agreed that with the frequent use of contractors and employment mobility within *The Bank*, adhering to a standard practice made sense:

> .. because you're working with people who move around a lot, obviously it's going to help a lot if everyone's using the same style. But I think, like anything, it's the role of a methodology anywhere.

A further example of systemic power is that for *The Bank* to gain quality certification it needed to demonstrate accepted industry-wide practices in terms of project management and systems development. A project manager agreed that with a mixture of skill sets within *The Bank* a common standard had advantages:

> I believe the bank wants to reach some kind of a maturity where there is a format and standards. And (the SDM] just provides some visibility as far as ensuring that the outcomes are doable.

These more visible disciplinary forces exerted power in the form of technical and institutional work practices on developers within *The Bank*. The origin of this power did not come from the business client but from the discipline of systems development itself.

To summarise Dimension 4, *The Bank* was forced to follow industry standards because of the mobile nature of workers in the IT profession. This example illustrates a source of power emanating from the IT industry, and imposed on *The Bank* and developers a requirement to comply with industry, national and global work practices. The essence of systemic power, in this case, in favour of the business client was the unconscious acceptance of values, traditions and culture of the discipline of systems development by the developer. In sum, the power of the *system* was intertwined into the fabric of *The Bank*'s IT division, ensuring that demands and challenges were never made by developers, and that the status quo was never challenged. Many developers when asked if they discussed the relative merits of the methodology with other colleagues said 'they did not' as the following excerpt illustrates:

> It's one of those things that you discuss when you're relatively new to the organisation but after that it's just accepted. You do it because it's part of the culture.

In summary, using Hardy's four dimensions of power, developers lose out to the client because the client controls the critical resources (funding); in the second, by being recipients only in the decision-making forum (the client signs-off, and controls the SDM so that developers can't replace aspects of it); in the third, by being unaware of the client unwittingly manipulating the meaning and perception of the SDM through the use of symbols and rituals; and in the fourth, developers comply with the status quo because power is embedded in the organisation and discipline of systems development that everyone takes for granted. Consequently, the institutionalisation of power in terms of routine organizational work practices embedded in the SDM thus benefited the client group.

DISCUSSION AND CONCLUSION

Through the application of Hardy's (1985; 1996) model of systemic power, we provided a perspective illustrating how the power of the *system* is able to frame the SDM's meaning. This finding in terms of SDM enactment is innovative and illustrates where power structures have become institutionalised and taken for granted, that developers unwittingly enact the SDM in ways that replicate development status quo. Our analysis also provides empirical insight into the importance and utility of meaning and systemic power which have been relatively under-utilised in studies of information systems development (Azad and Faraj, 2010).

Second, we argue that Hardy's (1985, 1996) notion of systemic power mirrors the concept of institution (DiMaggio & Powell, 1983; Scott, 2001) and offers further explanation of how the 'power of the system' operates as an institutional source of power. Our analysis showed that Hardy's 4th dimension, based on the concept of disciplinary power, is woven into the relations and discourse throughout the organisation and helped shape developers' beliefs about their roles. According to new institutional theorists (DeMaggio & Powell, 1983; Scott, 2001) this form of disciplinary power works through the ongoing routines and day-to-day work practices in organisations. These institutionalised routines become part of the socialisation process that influences not only what people do but also how they feel about what they do. In this way, power is built into and diffused through the system itself. The system and discipline of systems development ensures that imposed methodology enabled functions are translated into social and organisational facts i.e. the unquestioned normative structures of the organisation (sign-off, walk-through meetings, acting professionally, etc). In sum, both Hardy's concept of systemic power and new institutional theory are complimentary because they discuss the same concept of discipline as an institutional source of power, and both concur that these forces largely occur outside the awareness of developers and business clients.

To conclude, this paper has accomplished its aim. First, it developed a theoretical framework grounded in a model of unobtrusive power and the concept of disciplinary agency, and applied it in the analysis of a contemporary case organisation. Our analysis was strengthened by Hardy's (1985) model of unobtrusive power, allowing us to understand through the 'management of meaning' and 'systemic power' why there was cooperation from developers with the client in a scenario involving a conflict of interest. Given the lack of studies in our discipline that integrate different dimensions of power (Jasperson et al, 2002) rich insight that enhances our understanding and explanation is an appropriate and significant contribution (Gregor, 2006).

REFERENCES

Azad, B. and Faraj, S. "Social power and information technology implementation: a contentious framing lens", *Information Systems Journal*, 21, 2010, pp. 33-61.

Beath, C. and Orlikowski, W. "The Contradictory Structure of Systems Development Methodologies: deconstructing the IS-business client relationship in Information Engineering", *Information Systems Research* 5(4), 1994, pp. 350-377.

Clegg, S. *Frameworks of Power*, Sage, 1989.

Dhillon, G. "Dimensions of Power and IS Implementation", *Information & Management*, 41, 2004, pp. 635-644.

Dhillon, G. Caldeira, M. and Wenger, M. "Intentionality and power interplay in IS implementation: The case of an asset management firm", *Journal of Strategic Information Systems*, 2011, 20, pp. 438-448.

DiMaggio, P. and Powell, W. "The iron cage revisited: institutional isomorphism and collective rationality in organisational fields", *American Sociological Review*, 48(2), 1983, pp. 147-160.

Eisenhardt, K, "Building Theories from Case Study Research", *Academy of Management Review*, 14(4), 1989, pp. 532-550.

Fitzgerald, B. Russo, N. and Stolterman, E. *Information Systems Development: Methods in Action*, McGraw Hill, Berkshire, 2002.

Gregor, S. "The Nature of Theory in Information Systems", *MIS Quarterly,* 30(3), 2006, pp. 611-642.

Hardy, C. "The Nature of Unobtrusive Power" *Journal of Management Studies*, 22(4), 1985, pp. 384-399.

Hardy, C. "Understanding Power: Bringing about Strategic Change", *British Journal of Management*, 7, 1996, pp. s3-s16.

Hardy, C. and Leiba-O'Sullivan, S. "The Power Behind Empowerment: Implications for Research and Practice", *Human Relations*, 51(4), 1998, pp. 451-483.

Heiskanen, A., Newman, M. and Eklin, M. "Control, Trust, Power, and the Dynamics of Information System Outsourcing Relationships: A Process Study of Contractual Software Development", *Journal of Strategic Information Systems*, 17, 2008, pp. 268-286.

Hekkala, R. Urquhart, C., and Iivari, N. "Who is in Charge, and Whose Rules are Followed? Power in an Inter-Organisational Project", 17th European Conference on Information Systems, 2009.

Horton, K. "Strategy, practice, and the dynamics of power", *Journal of Business Research*, 56, 2003, pp. 121-126.

Howcroft, D. and Light, B. "Reflections on Issues of Power in Packaged Software Selection", *Information Systems Journal*, 16, 2006, pp. 215-235.

Howcroft, D. and McDonald, R. "An Ethnographic Study of IS Investment Appraisal", *International Journal of Technology and Human Interaction*, 3(3), 2007, pp. 69-86.

Howcroft, D. and Wilson, M. "Paradoxes of Participatory Practices: the Janus Role of the Systems Developer", *Information and Organization*, 13, 2003, pp. 1-24.

Huisman, M. and Iivari, J. "Deployment of Systems Development Methodologies: Perceptual Congruence Between IS Managers and Systems Developers", *Information and Management*, 43, 2006, pp. 29-49.

Hussain, Z. and Cornelius, N. "The Use of Domination and Legitimation in Information Systems Implementation", *Information Systems Journal*, 19, 2009, pp. 197-224.

Jasperson, J. Carte, T. Saunders, C. Butler, B. Croes, H. and Zheng, W. "Power and Information Technology Research: a Metatriangulation Review", *MIS Quarterly*, 26(4), 2002, pp. 397-459.

Kautz, K. Madsen, S. and Nørbjerg, J. "Persistent Problems and Practices in Information Systems Development", *Information Systems Journal*, 17, 2007, pp. 217-239.

Kautz, K. "Investigating the Design Process: Participatory Design in Agile Software Development", *Information Technology and People*, 24 (3), 2011, pp. 217-235.

Klein, H., and Myers, M. "A Set of Principals for Conducting and Evaluating Interpretive Field Studies in Information Systems", *MIS Quarterly*, 23(1), 1999, pp 67-94.

Lawrence, T., Winn, M., and Jennings, P. "The temporal dynamics of institutionalization", *Academy of Management Review*, 26, 2001, pp. 624-644.

Lukes, S. *Power: A Radical View*, 1st edition, Palgrave-Macmillan, London, 1974.

Madsen, S. Kautz, K. and Vidgen, R. "A Framework for Understanding How a Unique and Local IS Development Method Emerges in Practice", *European Journal of Information Systems*, 15, 2006, pp. 225-238.

Markus, M, and Bjørn-Andersen, N. "Power over business clients: its exercise by systems professionals", *Communications of the ACM*, 30(6), 1987, pp. 498-504.

Markus, L. and Mao, J-Y. "Participation in Development and Implementation – Updating an Old, Tired Concept for Today's IS Contexts, *Journal of the Association for Information Systems*, 5(11-12), 2004, pp. 514-544.

Nandhakumar, J. and Avison, D. "The Fiction of Methodical Development: a field study of information systems development", *Information Technology and People*, 12(2), 1999, pp. 176-191.

Pfeffer, J. *Power in Organizations*. Marshfield, MA: Pitman, 1981.

Pickering, A. *The Mangle of Practice*, University of Chicago Press, Chicago, 1995.

Sabherwal, R.and Grover, V. "A Taxonomy of Political Processes in Systems Development", *Information Systems Journal*, 20(5), 2010, pp. 419-447.

Sauer, C. and Lau, C. "Trying to Adopt Systems Development Methodologies – a Case-based Exploration of Business Business clients' Interests", *Information Systems Journal*, 7, 1997, pp. 255-275.

Scott, W. *Institutions and Organisations*, Thousand Oaks, CA, Sage, 2001.

Schwandt, T. *Dictionary of Qualitative Inquiry*, 2nd Ed, Sage, 2001.

Sillince, J. and Mouakket, S. "Varieties of Political Process During Systems Development", *Information Systems Research*, 8(4), 1997, pp. 368-397.

Silva, L. "Epistemological and Theoretical Challenges for Studying Power and Politics in Information Systems", *Information Systems Journal*, 17, 2007, pp.165-183.

Smith, H. "The Business client/Information Systems Relationship - A Study in Power and Attitudes", *Journal of Information Technology Management*, 1(2), 1990, pp. 9-23.

Walsham, G. "Interpretive Case Studies in IS Research: Nature and Method", *European Journal of Information Systems*, 4(2), 1995, pp.74-81.

COPYRIGHT

Rowlands & Kautz © 2013. The authors assign to ACIS and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to ACIS to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.